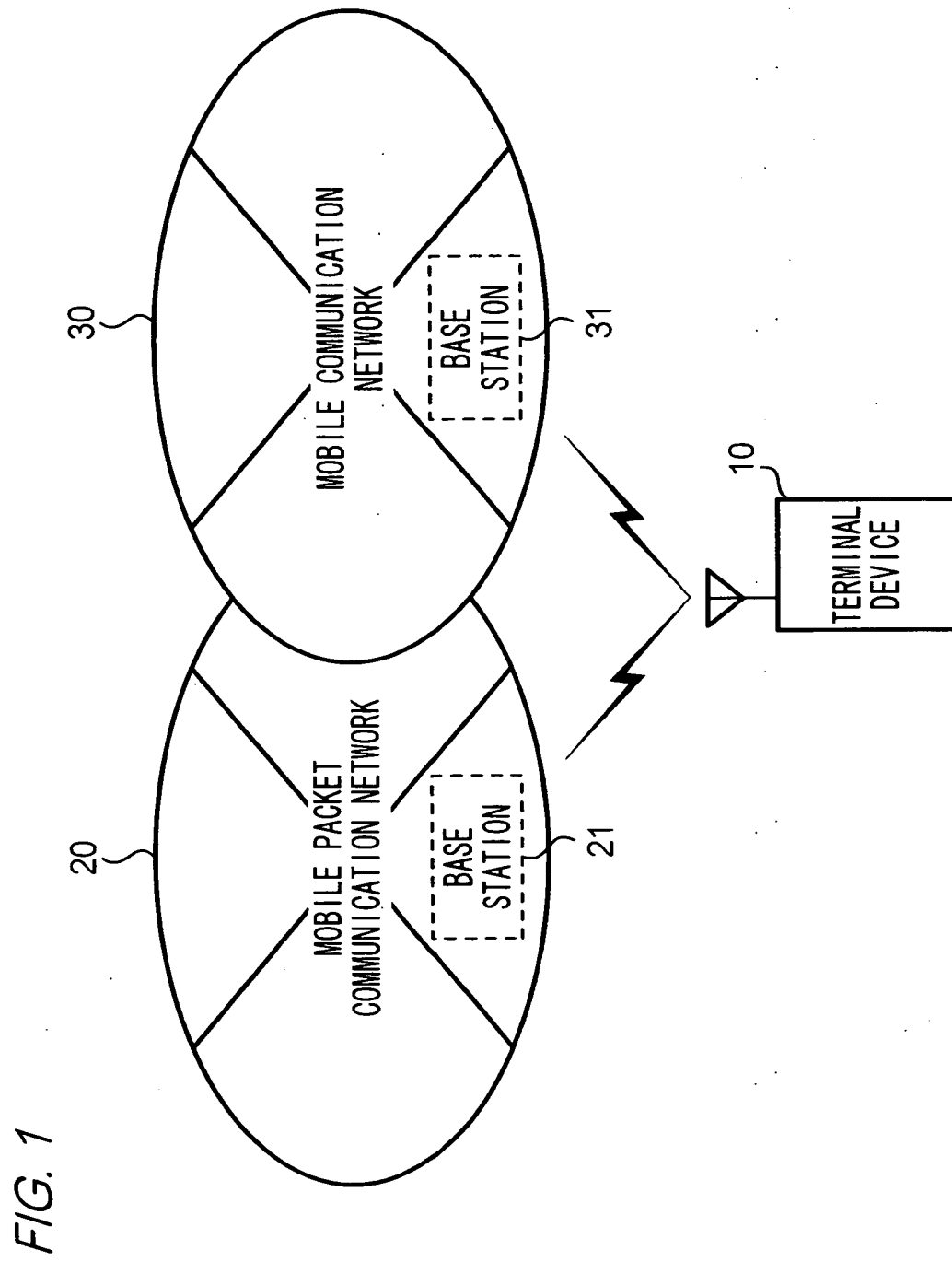


1/7



```

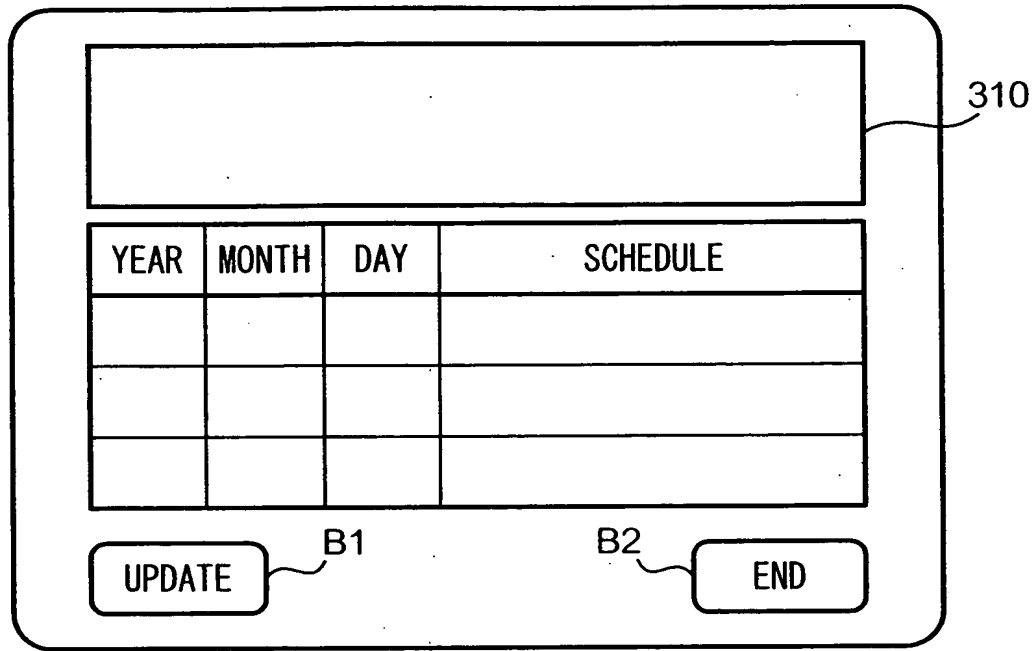
graph TD
    START([START]) --> SA1[DISPLAY SCHEDULE  
REGISTRATION SCREEN]
    SA1 --> SA2[ACQUIRE USER'S OPERATION]
    SA2 --> SA3{WHAT is USER'S  
OPERATION ?}
    SA3 -- "UPDATE BUTTON  
B1 is PRESSED" --> SA4[STORE SCHEDULE DATA]
    SA4 --> SA5[HIDE SCHEDULE  
REGISTRATION SCREEN]
    SA3 -- "END BUTTON B2  
is PRESSED." --> SA5
    SA3 -- "OTHERS" --> SA2
    SA5 --> END([END])
  
```

```
graph TD; START([START]) --> SB1[DISPLAY SCHEDULE  
REGISTRATION SCREEN AGAIN]; SB1 --> SB2[DISPLAY MESSAGE  
CORRESPONDING TO EVENT FLAG]; SB2 --> END([END]);
```

The flowchart illustrates the process for displaying a schedule registration screen. It begins with a terminal symbol labeled "START". An arrow points down to a rectangular process block labeled "DISPLAY SCHEDULE REGISTRATION SCREEN AGAIN", which is identified by the label "SB1" on its right side. Another arrow points down from this block to a second rectangular process block labeled "DISPLAY MESSAGE CORRESPONDING TO EVENT FLAG", identified by the label "SB2" on its right side. A final arrow points down from this block to a terminal symbol labeled "END".

3/7

*FIG. 3*

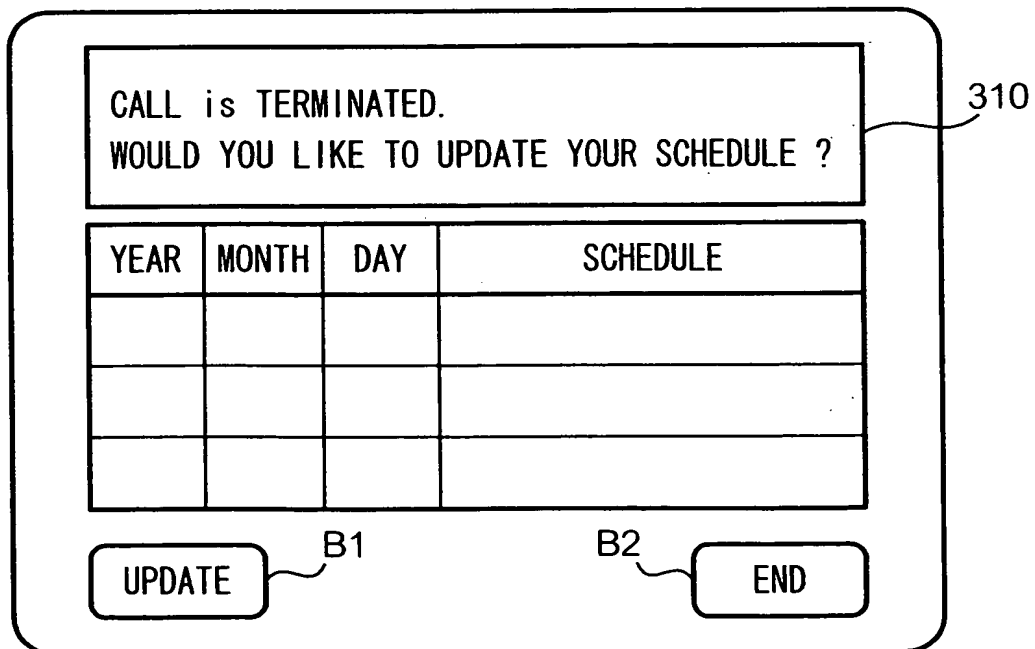


310

YEAR	MONTH	DAY	SCHEDULE

UPDATE B1 B2 END

*FIG. 9*



310

CALL is TERMINATED.  
WOULD YOU LIKE TO UPDATE YOUR SCHEDULE ?

YEAR	MONTH	DAY	SCHEDULE

UPDATE B1 B2 END

4/7

FIG. 5

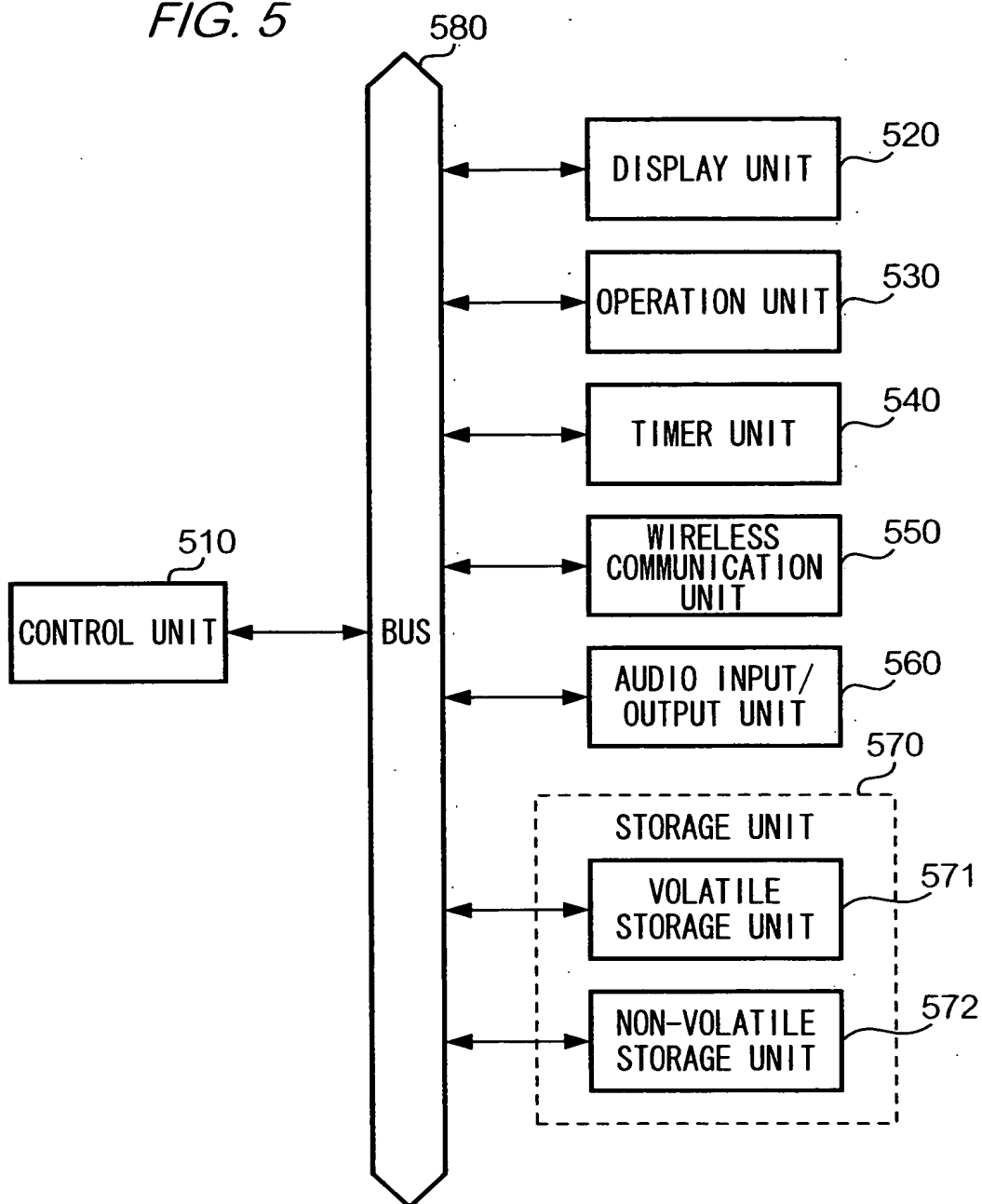


FIG. 6

PROGRAM IDENTIFIER	EVENT FLAG
SCHEDULER	1

5/7

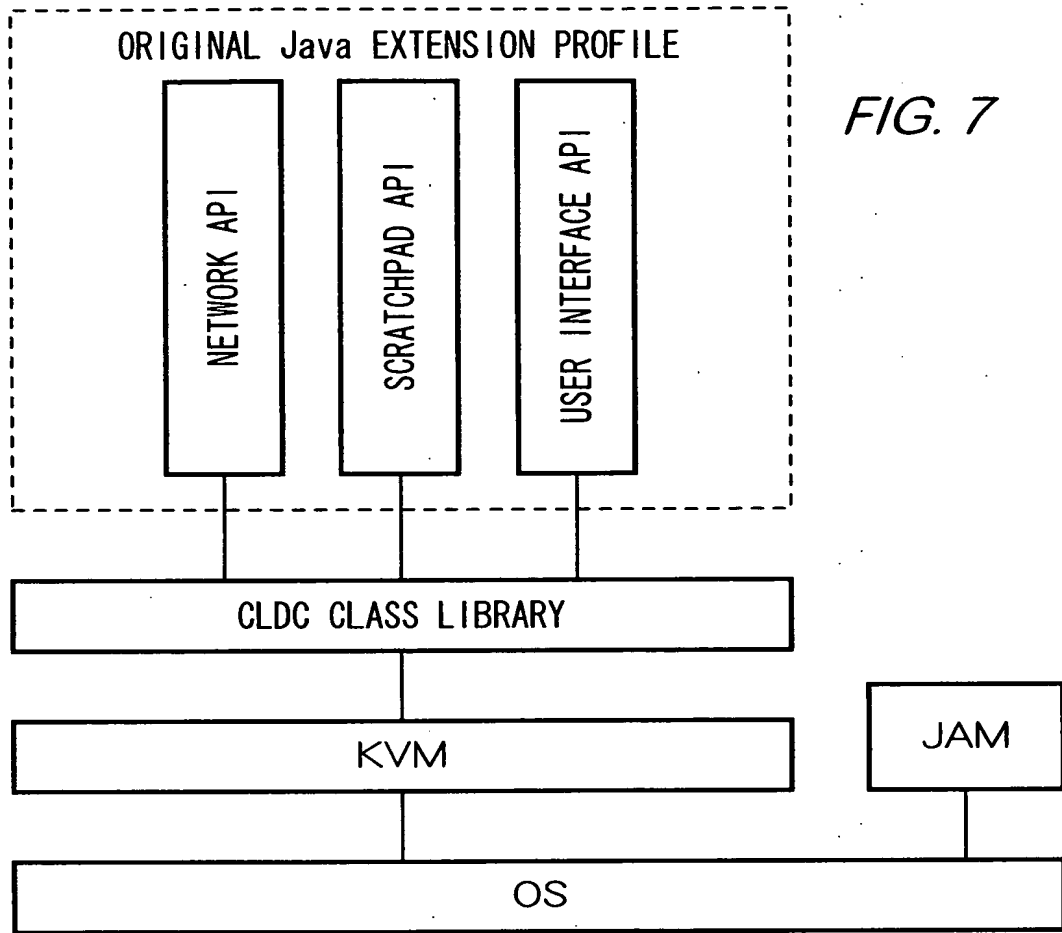
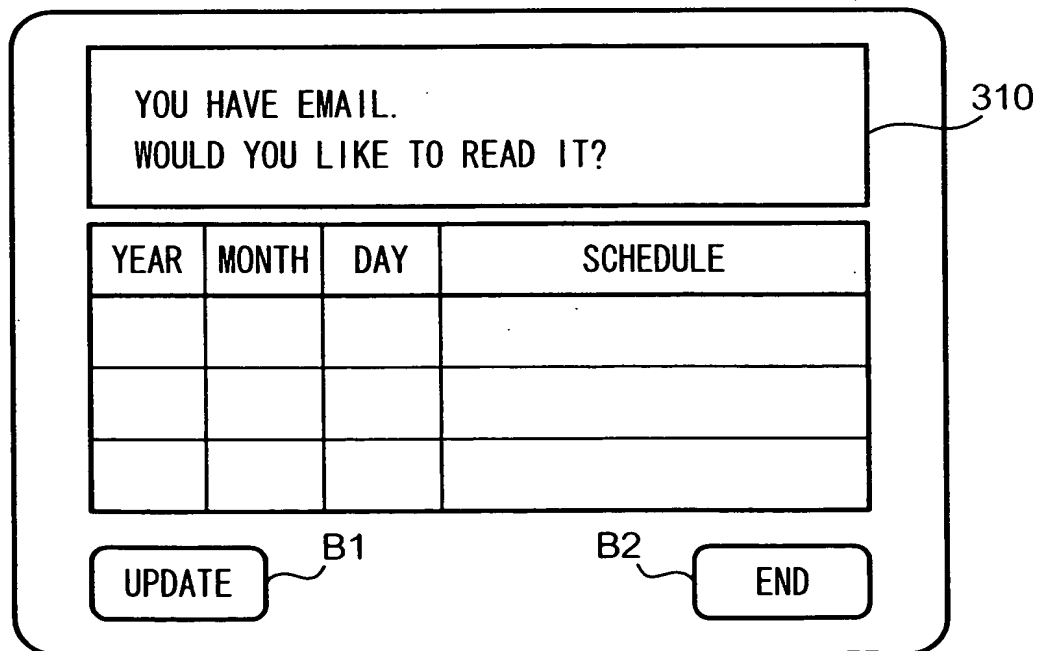
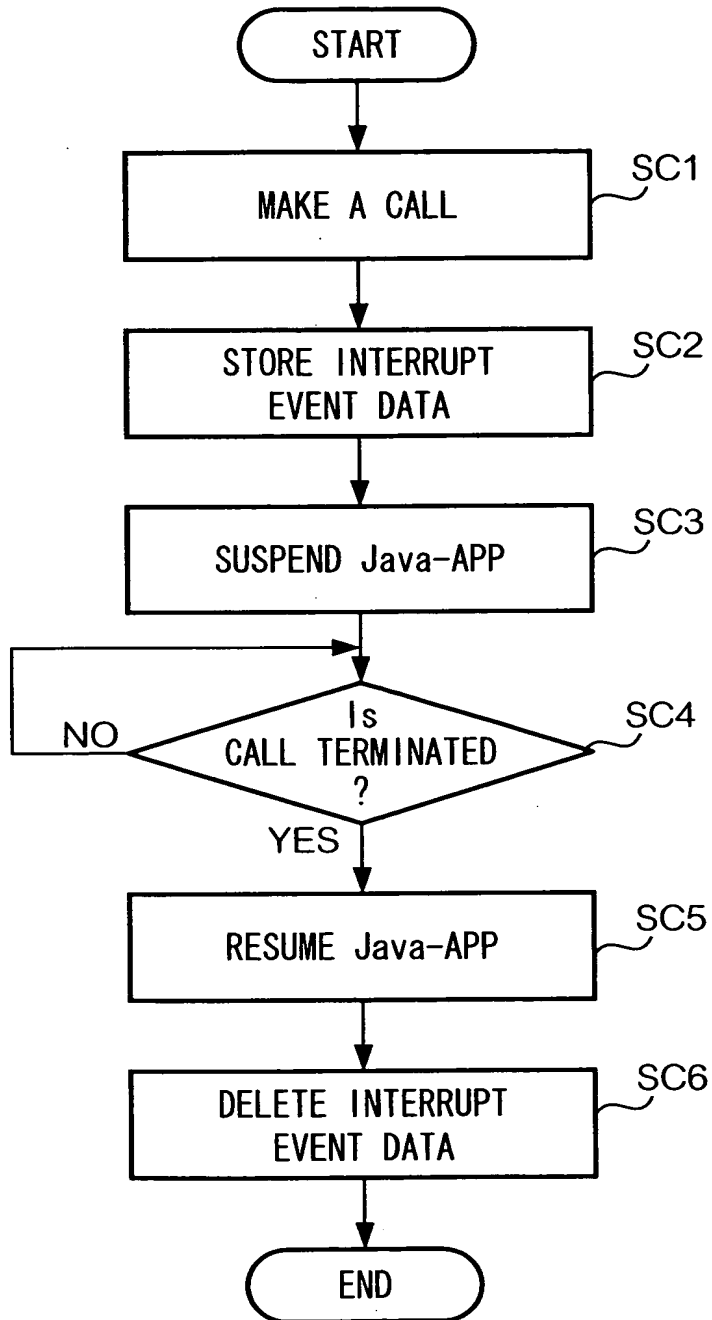


FIG. 11



6/7

FIG. 8



7/7

*FIG. 10*

